```
In[•]:= Rot[pd_PD] := Module[{n, xs, x, rots, Xp, Xm, front = {1}, k},
   n = Length@pd; rots = Table[0, {2 n}];
   xs = Cases[pd, x_X :> {Xp[x[[4]], x[[1]]] PositiveQ@x
                          Xm[x[[2]], x[[1]]]     True   ];
   For[k = 1, k ≤ 2 n, ++k,
    If[FreeQ[front, -k],
     front = Flatten@Replace[front, k → (xs /. {
             Xp[k, l_] | Xm[l_, k] :> {l + 1, k + 1, -l},
             Xp[l_, k] | Xm[k, l_] :> (++rots[[l]]; {-l, k + 1, l + 1}),
             _Xp | _Xm :> {}
            }), {1}],
     Cases[front, k | -k] /. {k, -k} :> --rots[[k]];
    ]
   ];
   {xs /. {Xp[i_, j_] :> {+1, i, j}, Xm[i_, j_] :> {-1, i, j}}, rots}];
Rot[K_] := Rot[PD[K]];
```

```
In[•]:= CF[ℰ_] := Module[{vs = Union@Cases[ℰ, g__, ∞], ps, c},
   Total[CoefficientRules[Expand[ℰ], vs] /. (ps_ → c_) :> Factor[c] (Times @@ vs^ps)]];
```

```
In[•]:= T3 = T1 T2;
```

```
In[•]:= R1[s_, i_, j_] =
  CF[s (1/2 - g3ii + T2^s g1ii g2ji - g1ii g2jj - (T2^s - 1) g2ji g3ii + 2 g2jj g3ii - (1 - T3^s) g2ji g3ji -
      g2ii g3jj - T2^s g2ji g3jj + g1ii g3jj + ((T1^s - 1) g1ji (T2^(2 s) g2ji - T2^s g2jj + T2^s g3jj) +
        (T3^s - 1) g3ji (1 - T2^s g1ii - (T1^s - 1) (T2^s + 1) g1ji + (T2^s - 2) g2jj + g2ij)) / (T2^s - 1))];
```

```
In[•]:= Θ[{s0_, i0_, j0_}, {s1_, i1_, j1_}] := CF[
  s1 (T1^s0 - 1) (T2^s1 - 1)^-1 (T3^s1 - 1) g1,j1,i0 g3,j0,i1 ((T2^s0 g2,i1,i0 - g2,i1,j0) - (T2^s0 g2,j1,i0 - g2,j1,j0))]
```

```
In[•]:= Γ1[φ_, k_] = -φ/2 + φ g3kk;
```

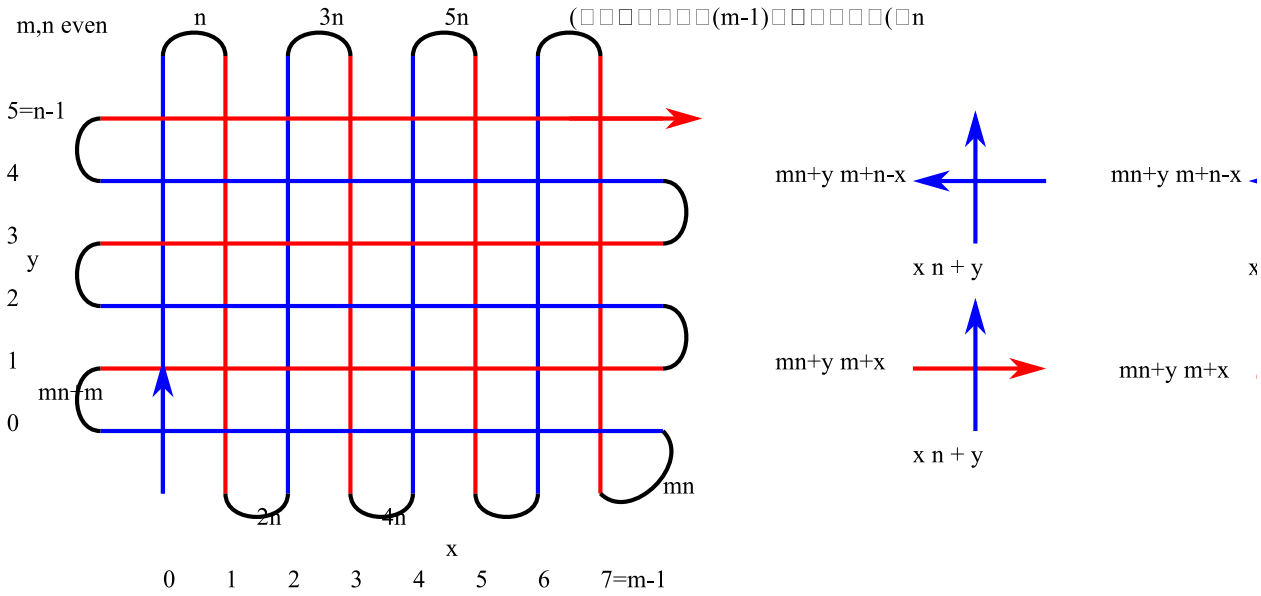
```
θ[K_] := Module[{Cs, φ, n, A, s, i, j, k, Δ, G, ν, α, β, gEval, c, z},
  {Cs, φ} = Rot[K]; n = Length[Cs];
  A = IdentityMatrix[2 n + 1];
  Cases[Cs, {s_, i_, j_} :> (A[[{i, j}, {i + 1, j + 1}]] += {{-T^s, T^s - 1}, {0, -1}})];
  Δ = T^((-Total[φ] - Total[Cs[[All, 1]]])/2) Det[A];
  G = Inverse[A];
  gEval[ℰ_] := Factor[ℰ /. g_ν_,α_,β_ :> (G[[α, β]] /. T -> T_ν)];
  z = gEval[Sum[Sum[θ[Cs[[k1]], Cs[[k2]]], {k2, 1, n}], {k1, 1, n}]];
  z += gEval[Sum[R_1 @@ Cs[[k]], {k, 1, n}]];
  z += gEval[Sum[Γ_1[φ[[k]], k], {k, 1, 2 n}]];
  {Δ, (Δ /. T -> T_1) (Δ /. T -> T_2) (Δ /. T -> T_3) z} // Factor];
```

```
PolyPlot[{Δ_, θ_}] := Module[{crs, m, m1, m2, maxc, minc, s, , rect, hex}, GraphicsColumn[{
    rect = {{0, 0}, {1, 0}, {1, 1}, {0, 1}};
    hex = Table[{Cos[α], Sin[α]} / Cos[2 π / 12] / 2, {α, 2 π / 12, 2 π, 2 π / 6}];
    If[Expand[Δ] === 0, Graphics[],
     crs = CoefficientRules[T^(m=-Exponent[Δ,T,Min]) Δ, {T}];
     maxc = N@Log@Max@Abs[Last /@ crs];
     minc = N@Log@Min@Select[Abs[Last /@ crs], # > 0 &];
     If[minc == maxc, s[_] = 0, s[c_] := s[c] = (maxc - Log@c) / (maxc - minc)];
     Graphics[crs /. ({x_} -> c_) :> {
         Lighter[Which[c == 0, White, c > 0, Red, c < 0, Blue], 0.88 s[Abs@c]],
         Polygon[({x + m - 1 / 2, 0} + #) & /@ rect] }, AspectRatio -> 1 / 5 ]
    ],
    If[Expand[θ] === 0, Graphics[{White, Disk[]}],
     crs = CoefficientRules[T_1^(m1=-Exponent[θ,T_1,Min]) T_2^(m2=-Exponent[θ,T_2,Min]) θ, {T_1, T_2}];
     maxc = N@Log@Max@Abs[Last /@ crs];
     minc = N@Log@Min@Select[Abs[Last /@ crs], # > 0 &];
     If[minc == maxc, s[_] = 0, s[c_] := s[c] = (maxc - Log@c) / (maxc - minc)];
     Graphics[{{White, Disk[{0, 0}, 1 + Cos[2 π / 12] Norm[{m1, m2}] / √2 ]},
       crs /. ({x1_, x2_} -> c_) :> {
           Lighter[Which[c == 0, White, c > 0, Red, c < 0, Blue], 0.88 s[Abs@c]],
           Polygon[({{1, -1/2}, {0, √3/2}}.{x1 - m1, x2 - m2} + #) & /@ hex] }
      }]
    ]
  }, Spacings -> 0]];
```

A mat-shaped knot based on an m by n rectangular grid. The integers m,n should be even and the

crossings can be chosen arbitrarily.

Probably all knots can be brought into this form.



In[•]:=

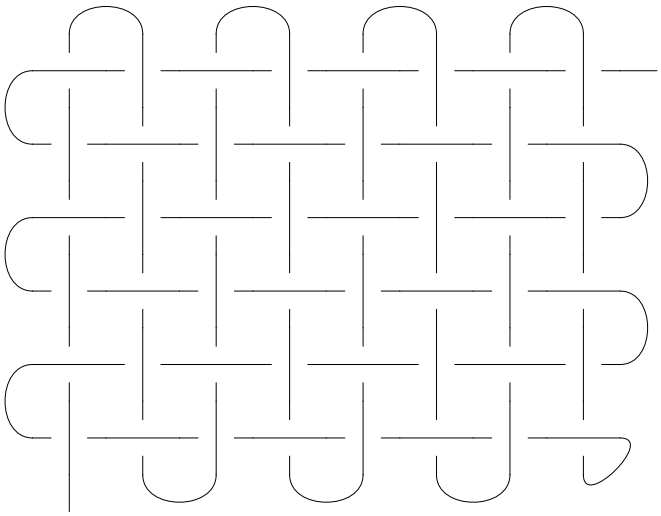
```
(*m,n, even, by default all xings positive. The
 list negxs flips the sign of those crossings.
  Generates PD code for the m by n mat with negative crossings indicated by negxs.*)
Mat[m_, n_, negxs_ : {}] := Module[{pd}, pd = (Flatten@Table[
        Switch[
         {EvenQ[x], EvenQ[y]},
         {True, True},
         X[m n + y m + m - x - 1, n x + y + 1, m n + y m + m - x, n x + y],
         {True, False},
         X[n x + y, m n + y m + x + 1, n x + y + 1, m n + y m + x],
         {False, True},
         X[n x + n - y - 1, m n + y m + m - x, n x + n - y, m n + y m + m - x - 1],
         {False, False},
         X[m n + y m + x, n x + n - y, m n + y m + x + 1, n x + n - y - 1]
        ]
        , {x, 0, m - 1}, {y, 0, n - 1}]
    ) /. {X[a_, b_, c_, d_] :> X[a + 1, b + 1, c + 1, d + 1]};
  Do[pd[[i]] = (pd[[i]] /. {X[a_, b_, c_, d_] :> X[d, a, b, c]}), {i, negxs}];
  PD @@ pd
 ]
(*Randomly assign signs to all crossings*)
RandMat[m_, n_] := Mat[m, n, RandomSample[Range[m n], RandomInteger[{0, m n}]]]
```

In[•]:=
```
DrawXing[x_, y_, cut_] :=
 If[cut === "v", {Line[{{x, y - 1/2}, {x, y - 1/4}}], Line[{{x, y + 1/4}, {x, y + 1/2}}],
   Line[{{x - 1/2, y}, {x + 1/2, y}}]}, {Line[{{x, y - 1/2}, {x, y + 1/2}}],
   Line[{{x - 1/2, y}, {x - 1/4, y}}], Line[{{x + 1/4, y}, {x + 1/2, y}}]}]
```

In[•]:=
```
DrawMat[m_, n_, negxs_ : {}] := Graphics[{
   Table[
    Switch[
     {EvenQ[x], EvenQ[y]},
     {True, True},
     DrawXing[x, y, If[MemberQ[negxs, y m + x + 1], "v", "h"]],
     {True, False},
     DrawXing[x, y, If[MemberQ[negxs, y m + x + 1], "h", "v"]],
     {False, True},
     DrawXing[x, y, If[MemberQ[negxs, y m + x + 1], "h", "v"]],
     {False, False},
     DrawXing[x, y, If[MemberQ[negxs, y m + x + 1], "v", "h"]]
    ]
    , {x, 0, m - 1}, {y, 0, n - 1}],
   Table[{Arrowheads[{{0, .7}}],
     Arrow[BezierCurve[{{x, -1/2}, {x, -1}, {x + 1, -1}, {x + 1, -1/2}}]]}, {x, 1, m - 3, 2}],
   Table[{Arrowheads[{{0, .7}}], Arrow[
      BezierCurve[{{x, n - 1/2}, {x, n}, {x + 1, n}, {x + 1, n - 1/2}}]]}, {x, 0, m - 2, 2}],
   Table[{Arrowheads[{{0, .7}}],
     Arrow[BezierCurve[{{-1/2, y}, {-1, y}, {-1, y + 1}, {-1/2, y + 1}}]]}, {y, 0, n - 2, 2}],
   Table[{Arrowheads[{{0, .7}}], Arrow[
      BezierCurve[{{m + -1/2, y}, {m, y}, {m, y + 1}, {m + -1/2, y + 1}}]]}, {y, 1, n - 3, 2}],
   BezierCurve[{{m - 1, -1/2}, {m - 1, -1}, {m, 0}, {m - 1/2, 0}}],
   {Arrowheads[{{0, 1}}], Arrow[{{m - 1/2, n - 1}, {m, n - 1}}]},
   {Arrowheads[{{0, 0.5}}], Arrow[{{0, -1}, {0, -1/2}}]}
  }]
DrawRandMat[m_, n_] := DrawMat[m, n, RandomSample[Range[m n], RandomInteger[m n]]]
```

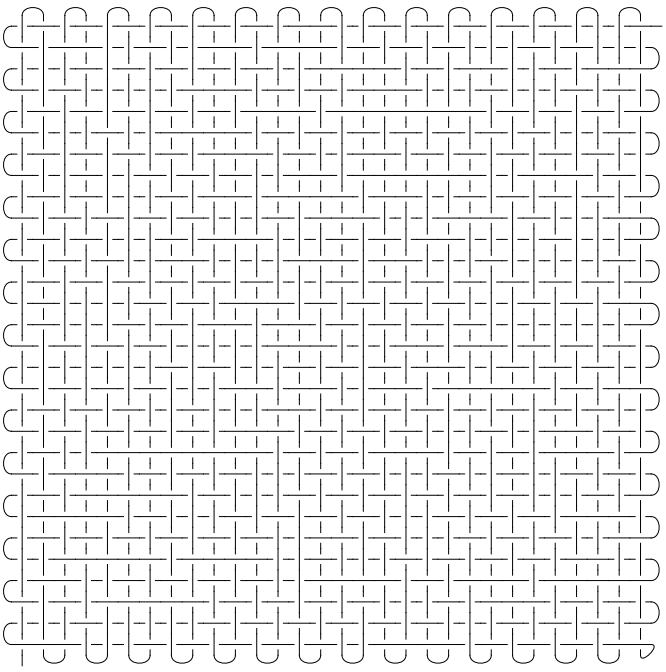
In[•]:=
```
DrawMat[8, 6]
```

Out[•]=



In[•]:=
```
DrawRandMat[30, 30]
```

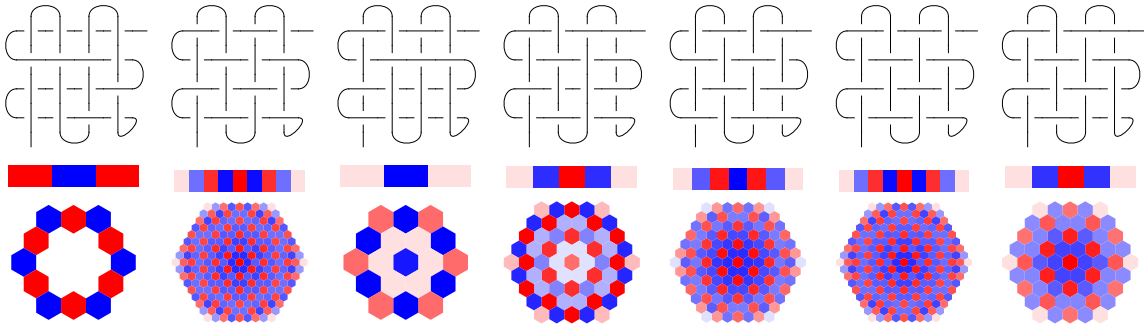
Out[•]=



In[•]:=
```
RandMatPair[m_, n_] := Module[{nxs = RandomSample[Range[m n], RandomInteger[m n]]},
  {DrawMat[m, n, nxs], PolyPlot@θ@Mat[m, n, nxs]}
 ]
```

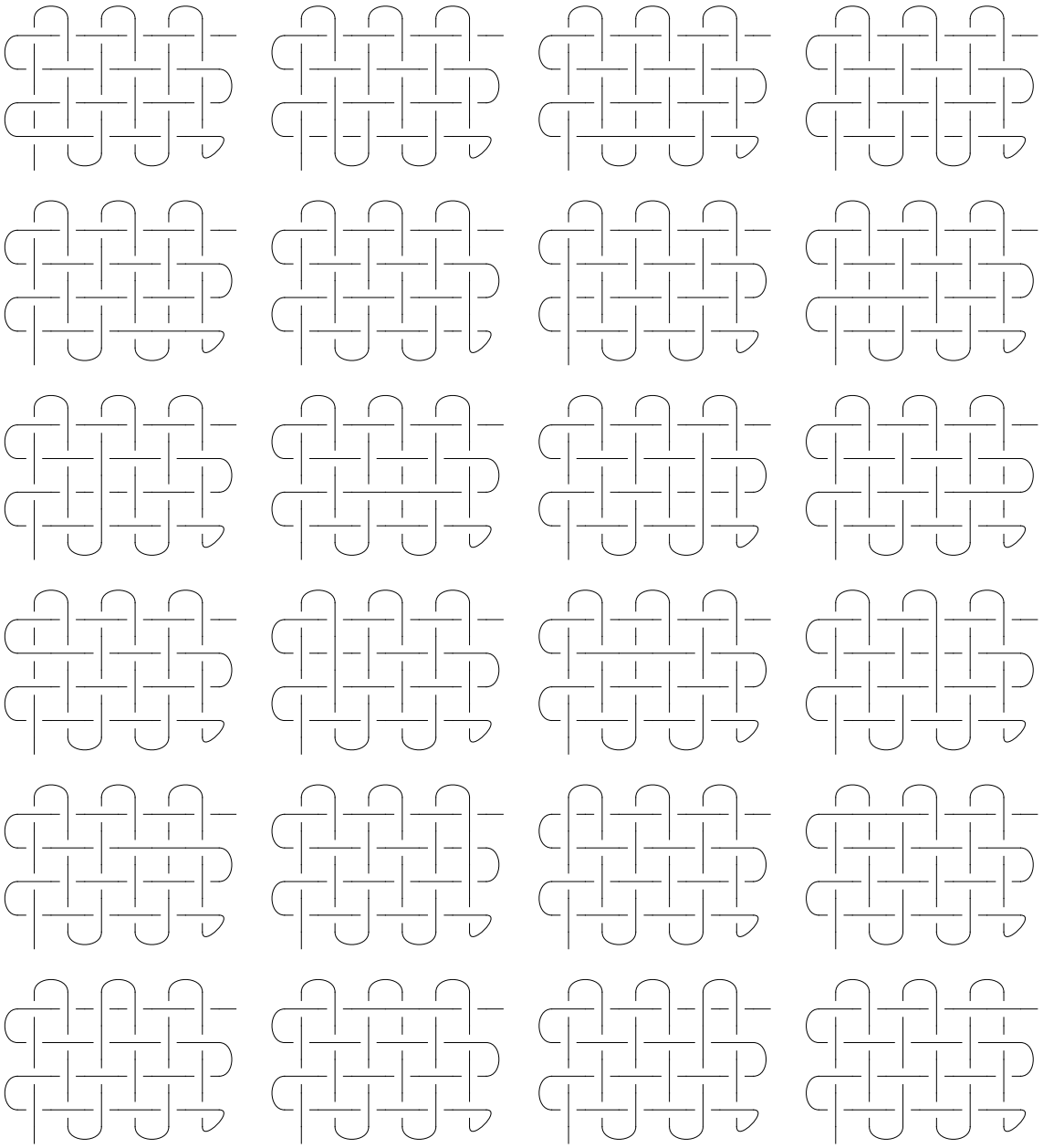
In[•]:=
```
GraphicsGrid[Table[RandMatPair[4, 4], {j, 7}] // Transpose]
```

Out[•]=



In[•]:=
```
Module[{m = 6, n = 4, μ, ν},
 GraphicsGrid[Table[
   DrawMat[m, n, {n μ + ν + 1}],
   {μ, 0, m - 1}, {ν, 0, n - 1}
  ]]
]
```

Out[•]=



```
In[•]:= Module[{m = 4, n = 4, μ, ν},
          GraphicsGrid[Table[
            K = Echo@Mat[m, n, {m μ + ν + 1}];
            PolyPlot[θ[K]],
            {μ, 0, m - 1}, {ν, 0, n - 1}
           ]]
        ]
```

» PD[X[1, 20, 2, 21], X[2, 22, 3, 21], X[28, 4, 29, 3], X[4, 30, 5, 29], X[8, 20, 9, 19], X[22, 8, 23, 7], X[6, 28, 7, 27], X[30, 6, 31, 5], X[18, 10, 19, 9], X[10, 24, 11, 23], X[26, 12, 27, 11], X[12, 32, 13, 31], X[16, 18, 17, 17], X[24, 16, 25, 15], X[14, 26, 15, 25], X[32, 14, 33, 13]]

Norm: The first Norm argument should be a scalar, vector, or matrix.

Dot: Nonrectangular tensor encountered.

Dot: Nonrectangular tensor encountered.

Dot: Nonrectangular tensor encountered.

General: Further output of Dot::rect will be suppressed during this calculation.

FilterRules: System`Private`SortOptions[$Failed] is not a valid replacement rule.

FilterRules: System`Private`SortOptions[$Failed] is not a valid replacement rule.

FilterRules:

FilterRules[System`Private`SortOptions[$Failed], {AlignmentPoint, AspectRatio, Axes, AxesLabel, AxesOrigin, AxesStyle, Background, BaselinePosition, BaseStyle, ColorOutput, ContentSelectable, CoordinatesToolOptions, DisplayFunction, Epilog, FormatType, Frame, FrameLabel, ≪5≫, ImageMargins, ImagePadding, ImageSize, ImageSizeRaw, LabelStyle, Method, PlotLabel, PlotRange, PlotRangeClipping, PlotRangePadding, PlotRegion, PreserveImageOptions, Prolog, RotateLabel, Ticks, TicksStyle}] is not a valid replacement rule.

General: Further output of FilterRules::rep will be suppressed during this calculation.

Lookup: The argument

FilterRules[FilterRules[System`Private`SortOptions[$Failed], {AlignmentPoint, AspectRatio, Axes, AxesLabel, AxesOrigin, AxesStyle, Background, BaselinePosition, BaseStyle, ColorOutput, ContentSelectable, CoordinatesToolOptions, DisplayFunction, Epilog, FormatType, Frame, ≪6≫, ImageMargins, ImagePadding, ImageSize, ImageSizeRaw, LabelStyle, Method, PlotLabel, PlotRange, PlotRangeClipping, PlotRangePadding, PlotRegion, PreserveImageOptions, Prolog, RotateLabel, Ticks, TicksStyle}], ≪1≫] is not a valid Association or a list of rules.

Lookup: The argument

FilterRules[FilterRules[System`Private`SortOptions[$Failed], {AlignmentPoint, AspectRatio, Axes, AxesLabel, AxesOrigin, AxesStyle, Background, BaselinePosition, BaseStyle, ColorOutput, ContentSelectable, CoordinatesToolOptions, DisplayFunction, Epilog, FormatType, Frame, ≪6≫, ImageMargins, ImagePadding, ImageSize, ImageSizeRaw, LabelStyle, Method, PlotLabel, PlotRange, PlotRangeClipping, PlotRangePadding, PlotRegion, PreserveImageOptions, Prolog, RotateLabel, Ticks, TicksStyle}], ≪1≫] is not a valid Association or a list of rules.

Lookup: The argument

FilterRules[FilterRules[System`Private`SortOptions[$Failed], {AlignmentPoint, AspectRatio, Axes, AxesLabel, AxesOrigin, AxesStyle, Background, BaselinePosition, BaseStyle, ColorOutput, ContentSelectable, CoordinatesToolOptions, DisplayFunction, Epilog, FormatType, Frame, ≪6≫, ImageMargins, ImagePadding, ImageSize, ImageSizeRaw, LabelStyle, Method, PlotLabel, PlotRange, PlotRangeClipping, PlotRangePadding, PlotRegion, PreserveImageOptions, Prolog, RotateLabel, Ticks, TicksStyle}], ≪1≫] is not a valid Association or a list of rules.

General: Further output of Lookup::invrl will be suppressed during this calculation.

Median: Rectangular array of real numbers is expected at position 1 in Median[{Lookup[System`Dump`opts, System`Dump`name], 1.}].

Median: Rectangular array of real numbers is expected at position 1 in Median[{System`Dump`opts, 360.}].

Part: Part specification {{Lookup[System`Dump`opts, System`Dump`name]}, {{{0., 1.}, {1., 0.}}}}[[All, All, 1, 1]] is longer than depth of object.

DeleteMissing: The argument {{Lookup[System`Dump`opts, System`Dump`name]}, {{{0., 1.}, {1., 0.}}}}[[All, All, 1, 1]] is not a valid Association or a list.

Part: Part specification {{Lookup[System`Dump`opts, System`Dump`name]}, {{{0., 1.}, {1., 0.}}}}[[All, All, 1, 2]] is longer than depth of object.

DeleteMissing: The argument {{Lookup[System`Dump`opts, System`Dump`name]}, {{{0., 1.}, {1., 0.}}}}[[All, All, 1, 2]] is not a valid Association or a list.

Part: Part specification {{Lookup[System`Dump`opts, System`Dump`name]}, {{{0., 1.}, {1., 0.}}}}[[All, All, 1, 1]] is longer than depth of object.

General: Further output of Part::partd will be suppressed during this calculation.

DeleteMissing: The argument {{Lookup[System`Dump`opts, System`Dump`name]}, {{{0., 1.}, {1., 0.}}}}[[All, All, 1, 1]] is not a valid Association or a list.

General: Further output of DeleteMissing::invrp will be suppressed during this calculation.

MapThread: Object Graphics`GraphicsGridDump`sizes$8292 at position {2, 1} in MapThread[Graphics`GraphicsGridDump`DividerCoordinates, {Graphics`GraphicsGridDump`sizes$8292, Graphics`GraphicsGridDump`ExpandOption[Spacings, {Automatic, 0}, {3, 2}, Graphics`GraphicsGridDump`sizes$8292]}] has only 0 of required 1 dimensions.

MapThread: Object −Graphics`GraphicsGridDump`sizes$8292 at position {2, 1} in MapThread[Graphics`GraphicsGridDump`DividerCoordinates, {−Graphics`GraphicsGridDump`sizes$8292, −Graphics`Graphic`.. sGridDump`ExpandOption[Spacings, {Automatic, 0}, {3, 2}, Graphics`GraphicsGridDump`sizes$8292]}] has only 0 of required 1 dimensions.

MapThread: Object Min[−Graphics`GraphicsGridDump`sizes$8292, −Graphics`GraphicsGridDump`ExpandOption[Spacings, {Automatic, 0}, {3, 2}, Graphics`GraphicsGridDump`sizes$8292]] at position {2, 1} in MapThread[{Graphics`GraphicsGridDump`DividerCoordinates, Graphics`GraphicsGridDump`DividerCoordinates}, {Min[−Gra`.. phics`GraphicsGridDump`sizes$8292, −Graphics`GraphicsGridDump`ExpandOption[Spacings, {Automatic, 0}, {3, 2}, Graphics`GraphicsGridDump`sizes$8292]], Max[−Graphics`GraphicsGridDump`sizes$8292, −Graphics`GraphicsGri`.. dDump`ExpandOption[Spacings, {Automatic, 0}, {3, 2}, Graphics`GraphicsGridDump`sizes$8292]]}] has only 0 of required 1 dimensions.

General: Further output of MapThread::mptd will be suppressed during this calculation.

```
» PD[X[20, 2, 21, 1], X[21, 2, 22, 3], X[28, 4, 29, 3], X[4, 30, 5, 29], X[8, 20, 9, 19], X[22, 8, 23, 7],
  X[6, 28, 7, 27], X[30, 6, 31, 5], X[18, 10, 19, 9], X[10, 24, 11, 23], X[26, 12, 27, 11],
  X[12, 32, 13, 31], X[16, 18, 17, 17], X[24, 16, 25, 15], X[14, 26, 15, 25], X[32, 14, 33, 13]]
```

Out[•]=

```
$Aborted
```

```
In[•]:= Θ@PD[X[1, 20, 2, 21], X[2, 22, 3, 21], X[28, 4, 29, 3], X[4, 30, 5, 29],
          X[8, 20, 9, 19], X[22, 8, 23, 7], X[6, 28, 7, 27], X[30, 6, 31, 5],
          X[18, 10, 19, 9], X[10, 24, 11, 23], X[26, 12, 27, 11], X[12, 32, 13, 31],
          X[16, 18, 17, 17], X[24, 16, 25, 15], X[14, 26, 15, 25], X[32, 14, 33, 13]]
```

Out[•]=

$$\left\{\left\{\left\{\left\{\frac{1}{T^{23}},\frac{1}{T^{153}},\frac{1}{T^{283/2}}\right\},T^{\cdots 1\cdots}\right\}\right\},\right.$$
$$\left\{\left\{\left\{\frac{1}{T_1^{46}T_2^{46}}\left(15+\Theta\left[\begin{array}{ll}\{1,3,28\} & \text{PositiveQ}[X[28,4,29,3]]\\ \{-1,4,28\} & \text{True}\end{array},\begin{array}{ll}\{1,3,28\} & \text{PositiveQ}[X[28,4,29,3]]\\ \{-1,4,28\} & \text{True}\end{array}\right]+\right.\right.\right.$$
$$\Theta\left[\begin{array}{ll}\{1,3,28\} & \text{PositiveQ}[X[28,4,29,3]]\\ \{-1,4,28\} & \text{True}\end{array},\begin{array}{ll}\{1,5,30\} & \text{PositiveQ}[X[30,6,31,5]]\\ \{-1,6,30\} & \text{True}\end{array}\right]+$$
$$\Theta\left[\begin{array}{ll}\{1,3,28\} & \text{PositiveQ}[X[28,4,29,3]]\\ \{-1,4,28\} & \text{True}\end{array},\begin{array}{ll}\{1,7,22\} & \text{PositiveQ}[X[22,8,23,7]]\\ \{-1,8,22\} & \text{True}\end{array}\right]+$$
$$\cdots 265 \cdots+R_1[\{\{\{1,25,14\},\text{PositiveQ}[X[14,26,15,25]]\}\},\{-1,26,14\}]+$$
$$R_1[\{\{\{1,27,6\},\text{PositiveQ}[X[6,28,7,27]]\}\},\{-1,28,6\}]+R_1[\{\{\{1,29,4\},\text{PositiveQ}[X[4,30,5,29]]\}\},$$
$$\left.\{-1,30,4\}]+R_1[\{\{\{1,31,12\},\text{PositiveQ}[X[12,32,13,31]]\}\},\{-1,32,12\}]\right),$$
$$\frac{15+\cdots 271 \cdots+R_1[\{\{\{1,31,12\},\text{PositiveQ}[X[12,32,13,31]]\}\},\{-1,32,12\}]}{T_1^{306}T_2^{306}},\frac{1}{T_1^{283/2}T_2^{283/2}(T_1T_2)^{283/2}}$$
$$\left(15+\Theta\left[\begin{array}{ll}\{1,3,28\} & \text{PositiveQ}[X[28,4,29,3]]\\ \{-1,4,28\} & \text{True}\end{array},\begin{array}{ll}\{1,3,28\} & \text{PositiveQ}[X[28,4,29,3]]\\ \{-1,4,28\} & \text{True}\end{array}\right]+\right.$$
$$\cdots 269 \cdots+R_1[\{\{\{1,29,4\},\text{PositiveQ}[X[4,30,5,29]]\}\},\{-1,30,4\}]+$$
$$\left.R_1[\{\{\{1,31,12\},\text{PositiveQ}[X[12,32,13,31]]\}\},\{-1,32,12\}]\right)\},T_1^{\cdots 1\cdots}T_2^{\cdots 1\cdots}$$
$$\left.\left.\left.\left(\cdots 1\cdots\right)^{\cdots 1\cdots}\left(15+\cdots 271\cdots+R_1[\{\{\{1,31,12\},\text{PositiveQ}[X[12,32,13,31]]\}\},\{-1,32,12\}]\right)\right\}\right\}\right\}$$

Full expression not available (original memory size: 1.3 MB)